

AMENDMENTS TO CLAIMS

Please amend the claims as follows (*wherein additions are shown by underlining and deletions are shown by strikethrough in amended claims*):

- B1
1. (Currently amended): A computer-implemented method, comprising:
disabling interrupts on a computer system for entering a state of hibernation;
preparing a first set of data for writing a first set of content of volatile memory to a
disk;
instructing a controller to write the first set of data asynchronously to a the disk;
polling, in intermittent polling operations, a status register to determine when the
write to the disk is complete; and
while between polling operations, preparing a second set of data for writing a
second set of content of volatile memory to the disk.
 2. (Original): The method of claim 1 wherein preparing the first set of data for
writing includes compressing the data.
 3. (Original): The method of claim 1 wherein preparing the first set of data for
writing includes segmenting the data.
 4. (Original): The method of claim 1 further comprising, calling at least one
driver to obtain information about write limitations of the disk.

B1
5. (Original): The method of claim 1 further comprising, reserving at least one special section of memory for preparing the first set of data for writing.

6. (Original): The method of claim 1 further comprising, calling a disk driver to obtain information about memory requirements of the driver.

7. (Original): The method of claim 6 further comprising, reserving at least one special section of memory for the disk driver based on the memory requirements thereof.

8. (Currently amended): A computer-implemented method, comprising:
disabling interrupts on a computer system;
instructing a controller to read data asynchronously from a disk to an input buffer;
polling, in intermittent polling operations, a status register to determine when the read from the disk is complete; and

while between polling operations, decompressing at least some data in the input buffer, wherein decompressing at least some of the data in the input buffer comprises, accessing the data including tokens comprising literal symbols and length, offset pairings, ensuring that space to hold a predetermined number of at least two output symbols is present in an output buffer, selecting a token from the input data, and when the token comprises a literal symbol, copying the literal symbol into the output buffer without checking whether the symbol will fit into the output buffer, or when the token comprises a length, offset pairing, copying at least the predetermined number of symbols into the output buffer without checking whether each symbol will fit into the output buffer.

B\

9. (Original): The method of claim 8 further comprising, loading at least one driver to enable the controller to read the data asynchronously from the disk.

10-11. (Cancelled).

12. (Original): A computer-implemented method, comprising,
accessing compressed input data including tokens comprising literal symbols and length, offset pairings;
ensuring that space to hold a predetermined number of at least two output symbols is present in an output buffer;
selecting a token from the input data; and
if the token comprises a literal symbol, copying the literal symbol into the output buffer without checking whether the symbol will fit into the output buffer; or
if the token comprises a length, offset pairing, copying at least the predetermined number of symbols into the output buffer without checking whether each symbol will fit into the output buffer.

13. (Original): The method of claim 12 wherein ensuring that space to hold a predetermined number of at least two output symbols is present includes, dividing space remaining in the output buffer by the predetermined number.

B1 14. (Original): The method of claim 12 wherein the token comprises a length, offset pairing and wherein copying a number of symbols comprises, copying the predetermined number regardless of an actual length value in the length, offset pairing, and adjusting a pointer based on the actual value.

15. (Original): The method of claim 12 further comprising, determining that space to hold a predetermined number of at least two output symbols may not be present in an output buffer, and invoking another decoder to decompress the input data into the output buffer.

16-21. (Cancelled).

22. (New): A computer-readable medium having computer-executable instructions, comprising:

disabling interrupts on a computer system while resuming operation from hibernation;

instructing a controller to asynchronously read data from a disk to an input buffer, the data representing contents of volatile memory previously stored on the disk from volatile memory;

polling, in intermittent polling operations, a status register to determine when the read from the disk is complete; and

while between polling operations, decompressing at least some data in the input buffer.

23. (New): The computer-readable medium of claim 22 further comprising, loading at least one driver to enable the controller to read the data asynchronously from the disk.

24. (New): In a computer system configured for entering hibernation, a method comprising:

loading a hibernation engine;

calling a compression engine for compressing at least some data to be written to a hard disk;

calling a driver with the hibernation engine;

instructing, with the driver, a first controller to write the data to a hard disc controller; and

instructing, with the first controller, the hard disc controller to write the data from the first controller to a specific location on the hard disk.

25. (New): The method of claim 24 further comprising allocating a hibernation-safe memory location with the hibernation engine.

26. (New): The method of claim 24 further comprising, calling at least one driver to obtain information about write limitations of the disk.

B1
27. (New): The method of claim 24 wherein the compression engine includes a compression output buffer.

28. (New): In a computer system configured for resuming from hibernation, a method comprising:

loading a hibernation engine for calling a driver;

calling the driver for reading in a data sequence;

instructing a hard disc controller to read the data sequence from a specific location on a hard disk to a first controller; and

instructing the first controller to read data from the hard disc controller to the driver.

29. (New): The method of claim 28 further comprising:

acquiring a hibernation file status; and

terminating the method when the hibernation file status is not marked as valid.

30. (New): The method of claim 28 further comprising calling a decompression engine for receiving compressed data from the driver.

31. (New): The method of claim 30 further comprising decompressing the received compressed data using a plurality of decoders.